



Using Threat Analysis To Design Secure Systems

Michael Howard
Program Manager, Security
Windows 2000 Team

Agenda

- A dose of reality
- A threat modeling strategy
 - STRIDE
- A short example
- Best practices

Reality!

- All products are susceptible to attack
- Vulnerabilities lead to threats, which lead to attacks
- Simply adding 'security' doesn't solve anything

Vulnerability

Threat



Asset



Vulnerability

Mitigation Techniques



Patrolled!
ggrrr!



Building Secure Systems

**"You cannot build secure
systems
unless you know the
threats
to which your are
susceptible"**

A Threat Modeling Strategy

- **Brainstorm about:**
 - Which assets need protecting?
 - What value are the assets?
 - What threats are the assets susceptible to?
 - Prioritize threats
 - How do you mitigate the threats?

STRIDE

- S - Spoofing user identity
- T - Tampering with data
- R - Reputability
- I - Information disclosure
- D - Denial of service
- E - Elevation of privilege

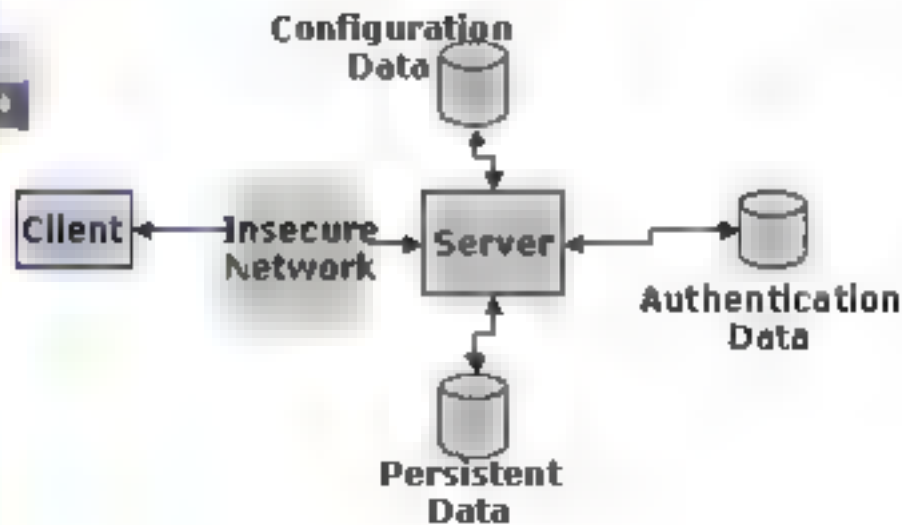
Assets

- Configuration data
- Authentication data
- Persistent data
- Data 'on the wire'
- State Data
- Temporary Data

Prioritization

- Chance of attack occurring
 - 1 = high, 10 = low
 - How much effort/cost/time is required to mount the attack?
- What's the cost/damage if attack occurs?
 - 1 = little, 10 = massive
- Risk = Damage/Chance
- Do high-risk items first

A Server Example 1



A Server Example 2

Configuration

DATA

Client

Insecure
Network

Server

Authentication
Data

Persistent
Data

SIRIDE and STRIDE

Attacker Intercepts data

Chance of occurring: 1

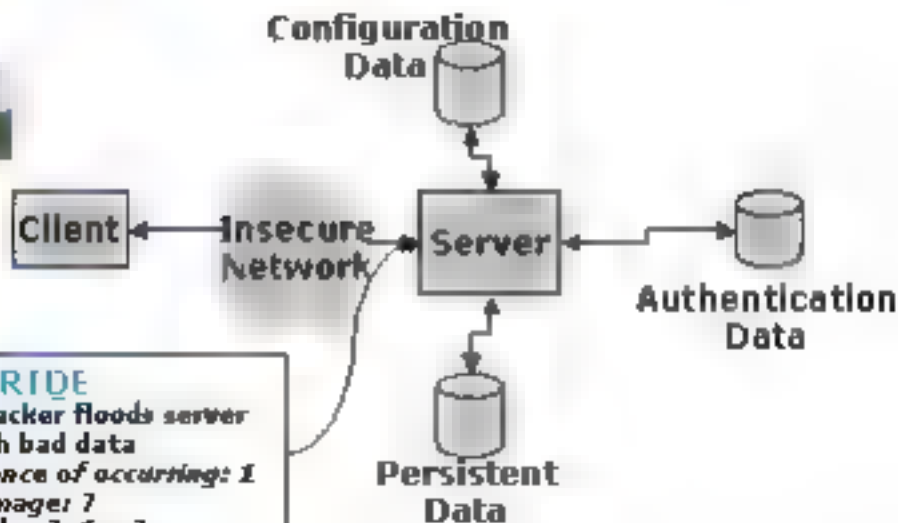
Damage: 10

Risk = 10 1 = 10

STRIDE normal data

STRIDE authn data

A Server Example 3



A Server Example 4

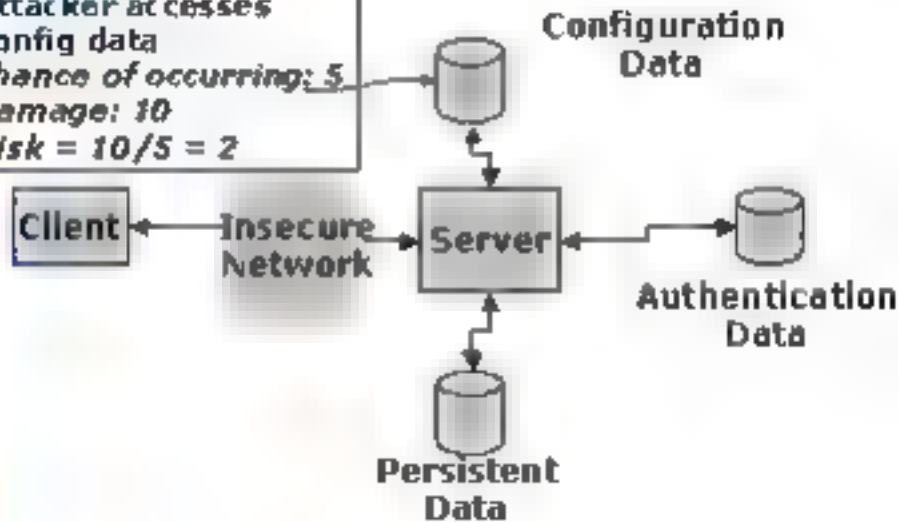
STRIDE

Attacker accesses
config data

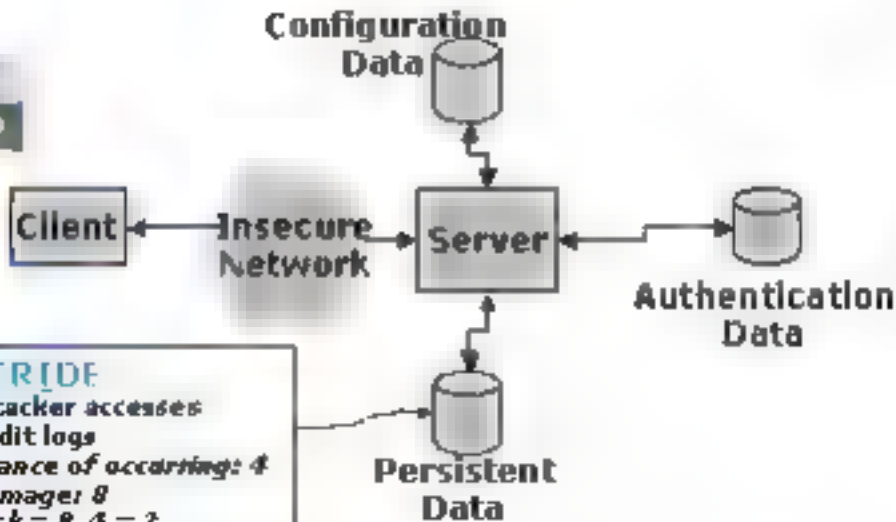
Chance of occurring: 5

Damage: 10

Risk = $10/5 = 2$



A Server Example 5



Mitigation Techniques

- S – strong authentication, don't store secrets
- T – hashes, digisigs, tamper resistant protocols
- R – digisigs, timestamps, secure logging
- I – strong access control mechanisms, encryption, don't store secrets
- D – filtering, throttling, QoS
- E – Run with least privilege

A Server Example 6

STRIDE and
STRIDE
Use SSL/TLS/IPSec
RPC with privacy/integrity

Client

Insecure
Network

Configuration
Data

STRIDE
ACLs
DigSig



Server



Authentication
Data



Persistent
Data

STRIDE
Packet filtering
Require Authn/Authz

STRIDE
ACLs
Auditing

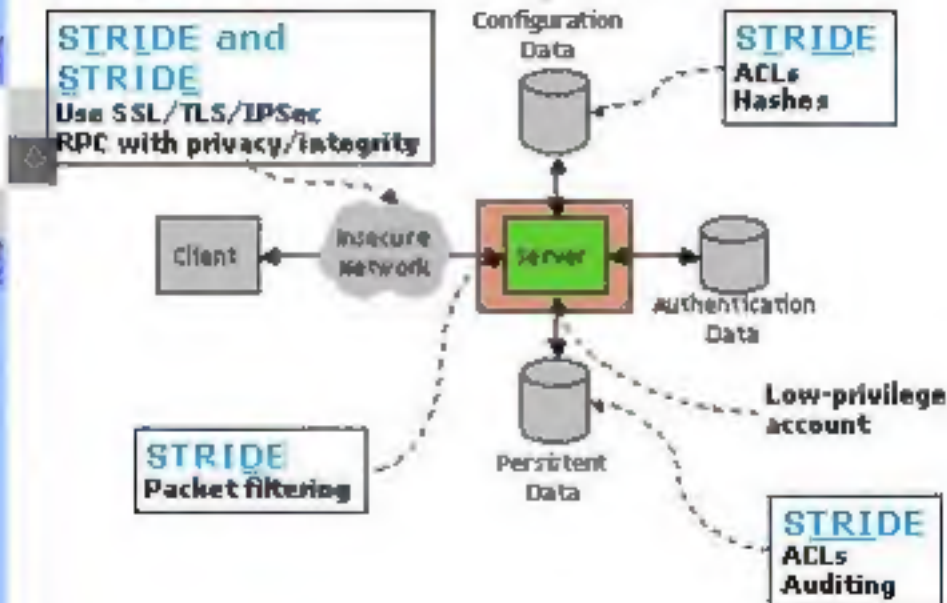
Development Best Practices

- Check for buffer overflows
- Don't run as Localsystem
- Do not require admin or power user
- Use negotiate, not NTLM SSP
- Request privacy/integrity on RPC/COM calls
- Use/robust MIDL flag

Web Development Best Practices

- Parse all user input using RegExp
 - <http://www.microsoft.com/technet/security/crssite.asp>
- Deny use of '..' in a filename
- Don't return physical paths to user

A Server Example 7



Shameless Plug!



ISBN: 0735609950

Summary

- A dose of reality
- A threat modeling strategy
 - STRIDE
- A short example
- Best practices